Xinyu Zhu. Building and Evaluating the Campus Online File Sharing Tool: UNC Dropbox System. A Master's paper for the M.S. in I.S. degree. April, 2011. 47 pages. Advisor: Paul M. Jones

In this project, a web-based online file sharing system called UNC Dropbox System was designed and implemented to provide UNC-CH affiliated personnel a free and secure file sharing platform. The system adopted the UNC Single Sign-on Service for authentication, which not only ensured the system security but also seamlessly integrated the system to the existing UNC web services.

A small group user testing with 12 participants from three UNC departments was conducted after system implementation. Participants were asked to test the system and submit a survey to evaluate the system from five aspects (content, accuracy, format, ease of use and timeliness). Survey answers were quantified to calculate user satisfaction scores on the performance of the system. The final results indicated that testing participants were generally satisfied with their experience with the system. User suggestions have also been collected during the system testing as important guidance for the future system improvements.

Headings:

Online File Sharing System

Web site – Design

Information System – Development

Information System – Evaluation

Surveys

BUILDING AND EVALUATING THE CAMPUS ONLINE FILE SHARING TOOL: UNC DROPBOX SYSTEM

by
Xinyu Zhu

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

April, 2011

Approved by:

_____

Advisor

Table of contents

Table of Figures

Table of Tables

Introduction

Nowadays, with the rapidly increasing number of digital devices that people possess and the booming of digital data usage in the daily life, needs for an effective way to share electronic data resources among multiple digital devices and between different people have become more and more urgent. Besides the traditional removable data storage devices and File Transfer Protocol (FTP) Servers, people are calling for a more convenient, platform independent and easy to use tool. The online file sharing system is designed to fulfill such urgent demand from people. It provides online personal space which allows users to create, upload, delete and share digital files with friends. Since user uploaded digital files are stored on the internet servers instead of any user's local machines and online file sharing systems always have web page user interface, people can easily access digital files from different digital devices through web browsers without the need to worry about synchronization and platform compatibility problem. The online file sharing system also makes sharing of digital files between people much easier. People can share files with their friends only by some clicks to select files and friends they want to share files with. The system will automatically notify their friends with necessary information to retrieve the file. Several internet service companies have already come out with their commercial online file sharing products and successfully attracted

lots of users. One typical example is Dropbox.com. Dropbox.com is founded on 2008 to provide both free and paid online file sharing service for users. Until Jan, 2010, they have already attracted over 4 million users (Wauters, 2010) and by June their revenue for 2010 has reached $14 million (Barret, 2010). Other similar online file sharing service providers include MEGAUPLOAD, RapidShare and 4Shared.com.

Although commercial online file sharing systems have already become quite mature and fulfilled most people's daily personal needs, it is inappropriate for organizations like research institutes, universities and government agencies, which are frequently dealing with confidential digital data, to use these systems for working purposes. Data security and confidentiality are two most significant concerns of these organizations. Therefore, in house development of online file sharing systems for internal usage has become one of the most popular solutions to deal with this dilemma. In academic field, University of Delaware (https://pandora.nss.udel.edu/), University of Essex (https://dropbox.essex.ac.uk/) and Lehigh University (https://dropbox.lehigh.edu/dropbox/) have all developed their own online file sharing systems called 'Dropbox service' for their affiliated members. These systems are securely protected by university authentication systems and electronic data are stored in central servers within the organizations.

University of North Carolina at Chapel Hill (UNC-CH), as one of the most well-known public universities in United States, at current stage, still does not have a centralized file sharing system available for its members. Under most circumstances,

digital files like course documents and research data are usually shared through email attachments and project, course or department based file sharing tools (Blackboard, AFS, department servers etc.). Large email attachments can easily wreak havoc on the email servers and severely influence the end users' internet experience. Uploading and downloading large email attachments is not only time consuming but also sometimes temporarily block the internet traffic (Schwartz, 2007). Size limit is another problem when using email attachment as a file sharing tool. 5 megabytes are usually too small for image or video data sharing. Project, course and department based file sharing tools perform much better than the email attachments in terms of file size limit and bandwidth consumption. However, they are only limited to use for particular courses and projects or limited to members of a particular department. Therefore, in this project, we were motivated to design and implement a free university-wide online file sharing system (UNC Dropbox System) available for all students, faculties and staffs in UNC-CH to use regardless of their course, project or department enrollment status. The online file sharing system and its data were designed to be securely protected by the university's authentication system. In order to test its reliability and usability, we have also conducted a small group user testing to evaluate user satisfactions on the system. Testing participants were all with professional knowledge on information system evaluation. A survey designed based on the End-User Computing Satisfaction instrument (Doll & Torkzadeh , 1988) was used to collect testing users' feedback. The evaluation results revealed that testing participants were overall satisfied with the performance of the UNC

Dropbox System. They also provided useful suggestions for the system future improvements.

System Characteristics and Structure

System Targeted Users and Functions

UNC Dropbox System is designed to serve two kinds of users, inside user and outside user. Inside users refers to people who are associated with UNC-CH including students, faculties and staffs while outside users are the rest of users on the internet. Inside users are allowed to use the UNC Dropbox System to share digital data with anyone, but outside users can only deliver files to inside users. The reason for opening the UNC Dropbox System to outside users with sharing restrictions is that many UNC projects are joint research with outside organizations. It is inevitable that UNC members need to share research materials with these outsiders and vice versa. Imposing restrictions on outside users do not block such communication and at the same time avoid unnecessary university resource consumption if outside users want to use the system for undesired purposes.

UNC Dropbox System has a web interface which can be accessed from any internet-connected digital devices with web browsers. The system interface is designed with ease of use as its top priority. Figure 1 is a screenshot of the system home page. From the figure, we can see that the user interface is quite clear and self-explanatory. All major functions that users were allowed to use were listed on this page in the form of

button menu. Every button is noted with a line of explanation text to indicate its functions. Red notification text is adopted on top of every page to show the login status of users. Authenticated users can login the systems to get more functions by clicking on the login button. It will re-direct users to a centralized authentication system (UNC Single Sign-on service) to get permissions to use the system. Such design ensures the security of the system. Moreover, one important feature of the UNC Single Sign-on authentication system is that UNC users who has logged in to one web service using this authentication system are also authenticated to use other campus web services using the same authentication system (e.g. blackboard, my.unc.edu etc.) without the need to login again, which means that the UNC Dropbox System has seamless integration with other campus web services. When authenticated users login the system, they get a different home page from the default with their names printed on the red welcome text (Figure 2).

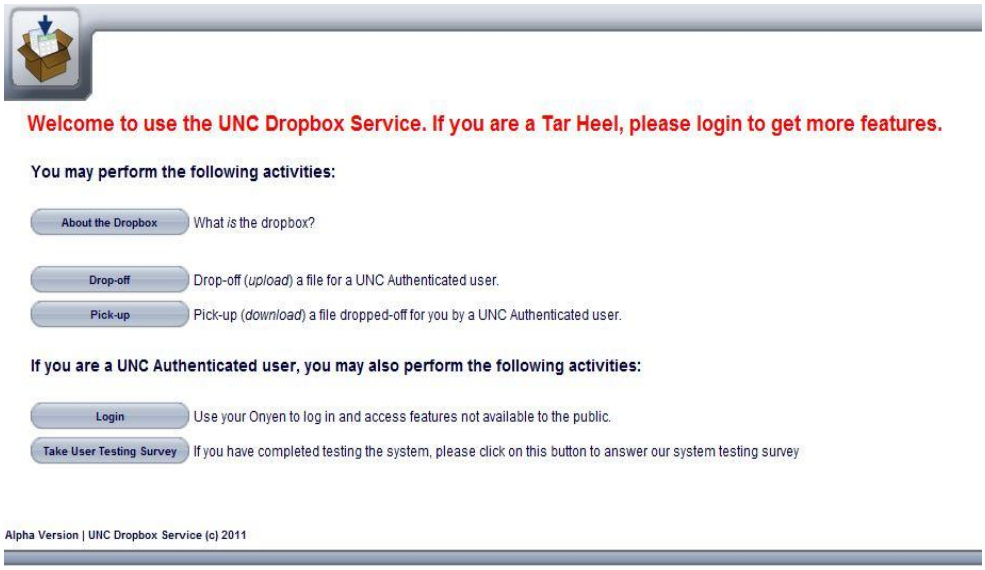

Figure 1: UNC Dropbox System Home Page

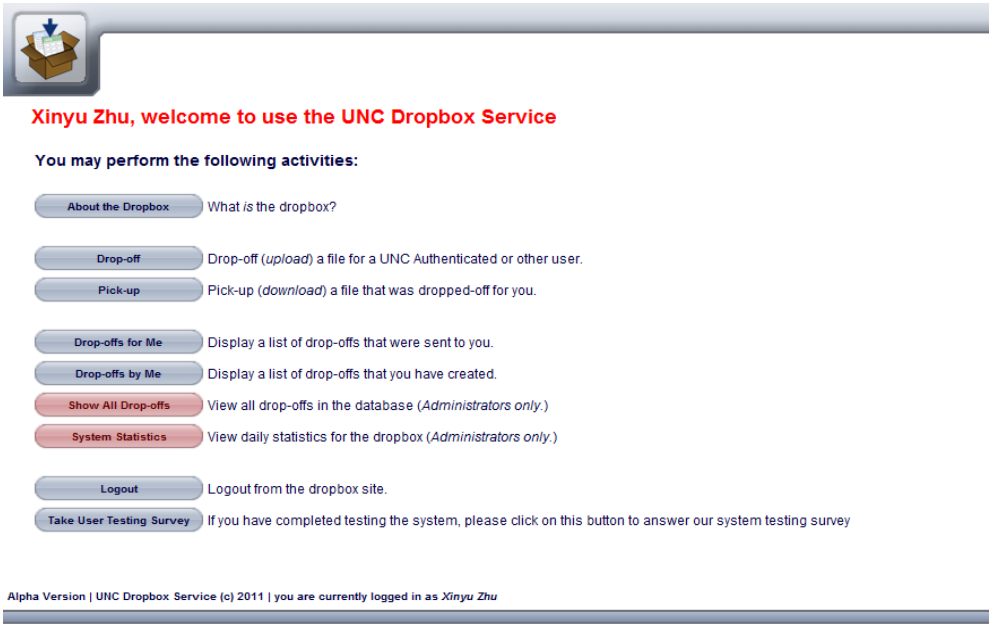

Figure 2: UNC Dropbox System Authenticated User Home Page

On the system home page, the first "About the Dropbox" button links users to a help document of the system. It introduces to users the motivations of designing the system, how to perform file sharing and pick-up using the system, and file uploading size limitations with different web browsers. The second "Drop-off" button brings users to the file sharing page (Figure 3), which is the key function of the system. In this page, users can provide the name and email addresses of recipients they want to share files with and upload the sharing files. They are allowed to share multiple files with multiple recipients at one time. When clicking on the "Drop-off the Files" button, an uploading progress bar will appear to indicate the file uploading progress and emails will be sent to recipients with necessary information to retrieve the sharing documents when file upload completes. The third button "Pick-up" on the system home page is used for people to pick up sharing files using the "Claim ID" and "Claim Pass code" provided in the system generated

notification email. With correct “Claim ID” and “Claim Pass Code” entered, the system displays a summary of pick-up documents to users (Figure 4) and users can download the files by simply clicking on the file name. When users download a sharing file from the UNC Dropbox System, a notification email will also be sent to inform the original sharer.

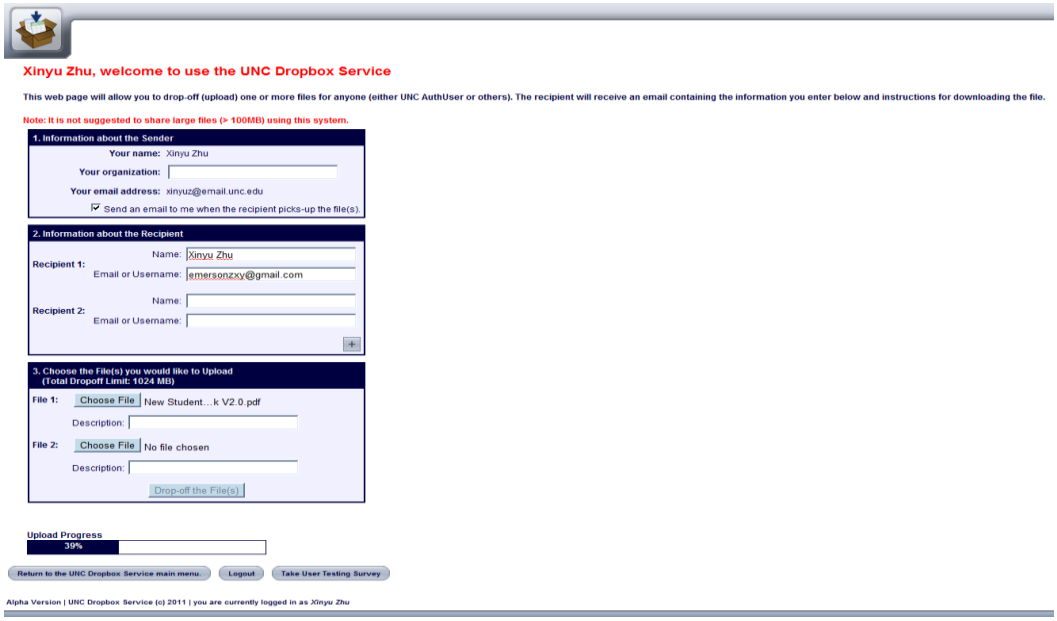

Figure 3: Drop-off page

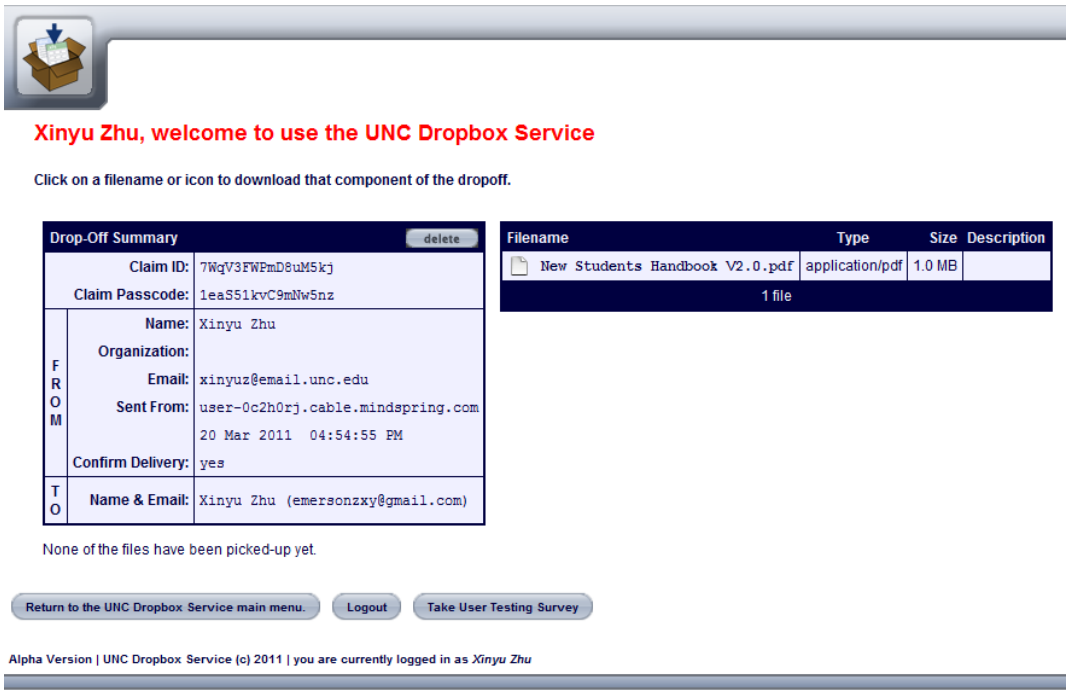

Figure 4: File Pick-up page

The fourth and fifth buttons ("Drop-offs for me" and "Drop-offs by me") on the system home page are functions only available for authenticated users. They provide users ways to track the list of files they received from other people and the files they shared with others. Users can also delete these files by clicking on the "delete" button in the drop-off summary. The sixth and seventh buttons ("Show All Drop-offs" and "System Statistics") are administrative functions only available to the system administrators. They are used to manage all the sharing files in the system and monitor the system usage status.

System Structure

UNC Dropbox System adopted a typical three tier information system design. As shown in Figure 5. We can see that end users with their web browsers sit at the

presentation tier. They use the web interface of the system to prepare their service requests. When they submit their requests, PHP scripts running on the application server at the Application Tier will process these requests. If they want to login to the system, PHP scripts will re-direct them to the UNC Shibboleth Server, which is the Single Sign-on Service provider, for authentication. Once the Application Tier receives the authentication results from the authentication server, it returns corresponding service pages to end users. If users request to share files with other people, PHP scripts at the Application Tier will validate their request based on authentication status and their submitted input. Once the request passes the validation check, PHP scripts will encrypt the sharing files, store them to the server file systems and insert file metadata records into the database system. If users request to pick up files, PHP scripts will validate the “Claim ID” and “Claim Pass Code” they provide and retrieve corresponding data from both the database system and the file system and send it back to the presentation tier.

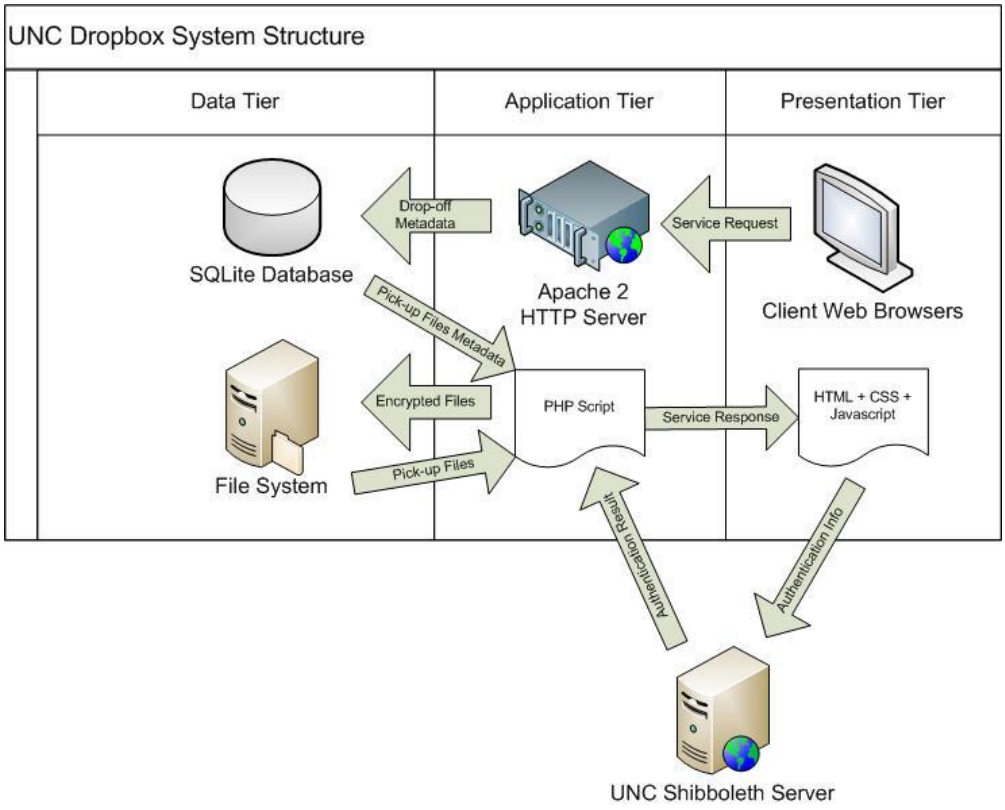

Figure 5: UNC Dropbox System Structure

Technology Overview

Linux server (Ubuntu 10.04 LTS)

In this project, Linux server running Ubuntu 10.04 LTS operating system was used to host the UNC Dropbox System. There are four reasons why I choose this server. First, Linux server is more secure than a Windows server at current point of time. Most viruses on the internet are designed to attack Windows machines since most people use it. Second, Ubuntu 10.04 LTS is free and a Long Term Support version. Ubuntu will provide up to five years free security updates for the operating system. Third, Ubuntu got high reputation on building Linux system and has a large user community, which means I

can easily find out solutions online when encountering technical difficulties. Last, I have past experience on using Ubuntu Linux operating system which made me easier to conduct server configuration.

Apache 2

Apache 2 is one of the most popular open source web server solutions. It is well known for its support on the server side programming language such as PHP Python and Perl. It is also one of the key components of the LAMP (Linux OS, Apache, MySql data base and PHP) free, open source software solutions to build a viable general purpose web server. According to a web server survey, as of January 2011, Apache HTTP server is hosting more than 59.13% of websites on the internet and over 66.62% of the million busiest (January 2011 Web Server Survey, 2011). In this project, the UNC Dropbox System is written purely in PHP5 and running on a Linux server. Apache 2 is therefore the best partner to collaborate with them and provide the system a stable and cost-effective operation environment.

Secure Socket Layer (SSL)

Secure Socket Layer is a cryptographic protocol that enforces communication security over the internet. It has been widely used to protect internet communications involving sensitive data exchange. The UNC Dropbox System is an internal online file sharing system for UNC-CH. It is inevitable to be used to share confidential research data between parties. Implementing SSL on the server hosting the UNC Dropbox System is a

necessity to prevent potential Man-in-the-middle attacks and enhance the security of the system.

Open Source UD Dropbox System in PHP5

In this project, I decided to build the UNC Dropbox System based on an existing open source Dropbox System developed by University of Delaware instead of building from scratch because of the following three reasons. First is time restriction. This project is supposed to be completed within two and half months. The schedule is too tight to build a new system from scratch and test it. Using a well developed open source system and modifying it to make it cater to the system requirements is a better choice to complete the project on time. The second reason is that this open source Dropbox system is developed purely on PHP5, which makes modifications on the source code much easier and it fulfills our platform independent with web user interface requirement on the Dropbox system. The file sharing functions it provides are also sufficient to meet the project expectations. The last reason is that this open source system has already been used in several other universities (University of Delaware, Florida States University etc.) for years, which proves its reliability.

Based on the original system, I put my efforts to customize it from two aspects. One is the authentication method. The original Dropbox system uses Lightweight Directory Access Protocol (LDAP) authentication, which, in fact, is also a secure authentication method to protect the system. However, it does not meet the system requirement of

integrating the UNC Dropbox System to other existing UNC web services through Single Sign-on service. Therefore, I disabled the LDAP authentication module in the system and implemented a new authentication module (Shibboleth authentication) to make it work appropriately with the UNC Single Sign-on authentication system. The other modification effort I did to the original system is the user interface upgrade. The user interface in the original design is not so user friendly and did not provide enough information for users to know their status and matters that need attention when using the system. According to this fact, I modified several parts of the system appearance and implemented some gadgets like the file uploading progress bar to enhance the usability of the system.

Postfix

One important feature of the UNC Dropbox System is the automatic notification email delivery to recipients when a drop-off request has been submitted. To realize this function, a mail transfer agent (MTA) is necessary to be installed on the server. Postfix is a free, open source MTA that has been widely used as a fast, easy-to-administer and secure alternative to the Sendmail MTA. With appropriate configurations, it enables the PHP scripts to send emails using mail() function through Postfix MTA on the hosting server. Therefore, it has been adopted in the UNC Dropbox System to support its automatic mail delivery feature.

PHP APC Package

Alternative PHP Cache (APC) is a free, open source framework which optimizes the PHP intermediate code and caches data point in the shared memory. PHP 5.2 with APC package installed enables PHP scripts to hook data points that are cached during file transfer progress. When a PHP script receives a file upload request, the interpreter automatically checks the $_POST array for a hidden variable called APC_UPLOAD_PROGRESS, which becomes a cached variable storing the information about the uploading progress and can be accessed by the script. With periodically reading this variable and updating the web page content using javascript, real time file uploading progress monitor can be realized to enhance the usability of web applications. In this project, I used this method to implement a file uploading progress bar in the UNC Dropbox System, which eliminated the problem that users have no idea of the uploading status when they are sharing large files with others.

Shibboleth

Shibboleth is a powerful alternative to LDAP for user management and authentication because of its single-sign-on functionality. Unlike the LDAP authentication model in which the application server performs the authentication process and the LDAP server only provides the hash for the application server to map user provided login information to the existing records, in the Shibboleth authentication model the Single Sign-on server is the only place handling the authentication process. Once a user login to a system via the Shibboleth authentication service provided by a Single Sign-on server, all other systems using the same authentication method also get informed

that this user has been authenticated. This user will not need to login again when he/she try to access other systems before his/her login session expires. Shibboleth authentication provides a seamless integration solution for all web applications in a system.

In UNC-CH, currently many services like connectCarolina and Blackboard have already implemented Shibboleth authentication using the service provided by the UNC Single Sign-on server. In this project, in order to make the UNC Dropbox system better integrate to the existing campus web services and offer end users more convenience, the Shibboleth authentication was implemented on the UNC Dropbox system too.

System Evaluation Method Overview

Measurement of Information System Success

How to measure the success of an information system has attracted researchers' interests for a long time. Since 1970s, several important criteria have been identified to evaluate the success of information systems as follows: system usage (Igbaria, Pavri, & Huff, 1989; Lucas, 1978), user satisfaction (Baroudi & Orlikowski, 1988; Powers & Dickson, 1973; S.Rivard & Huff, 1988) and performance (Lucas, 1978). Although system usage has once been used extensively by some researchers (Ein-Dor & Segev, 1982; Igbaria, Pavri, & Huff, 1989; Lucas, 1978), others (Ginzberg, 1978) criticized it as not an indicator strongly related to quality decision making. The basic assumption which made system usage an indicator of information system success is that users are free to choose information systems upon their requirements (Pikkarainen, Pikkarainen, Karjaluoto, &

Pahnila, 2006). If users use information systems on an involuntary basis, system usage is not able to reflect the true image (Doll & Torkzadeh, 1988). Thus, many researchers (Ginzberg, 1978; Tait & Vessey, 1988) recommended user satisfaction as a better alternative to evaluate information systems success. Ives et al. (1983) and Doll and Torkzadeh (1988) also explored that unlike system usage, in both voluntary and involuntary information system usage situation, user satisfaction was able to describe the success of information systems. In case of voluntary system usage, user satisfaction can also directly lead to the increase of system usage while in case of involuntary system usage, user satisfaction can still affect users willingness to use systems. Pikkarainen et al. (2006) summarized three reasons why user satisfaction became one of the most widely used measures of information system success. First, using satisfaction as a success factor makes common sense. Second, reliable tools are available for measuring user satisfaction. Last, other information system success measures are either weak conceptually or difficult to acquire.

The above summary of past research has clearly indicated that user satisfaction is a good indicator to evaluate the success of an information system in almost all situations while system usage is effective only under restricted conditions. Therefore, in this project, I adopted user satisfaction as the only indicator to evaluate the success of the system.

End User Computing Satisfaction Model

In the past, end user computing has once been defined in a broader way which includes three layers of end users: “indirect” users who use computers through other people, “intermediate” users who specify information output they need; and “direct” users who directly interact with the systems (Lefkovits, 1979). However, with the technology improvement, boundaries between different layers of end users have become more and more obscure. The prevalence of personal computer (PC) and the rapid development of application user interface have largely increased people’s computer skills and made information system usage much easier. Nowadays, the first layer of end users have gradually diminished and system end users commonly play a combined role of ‘intermediate’ and ‘direct’ end users. Therefore, in most End User Computing research, researchers tended to define end users as people with information needs who directly interact with the computer systems (Martin, 1982; McLean, 1979; Rockart & Flannery, 1983; Doll & Torkzadeh, 1988; Igbaria & Nachman, 1990; Abdinnour-Helm, Chaparro, & Farmer, 2005; Pikkarainen, Pikkarainen, Karjaluoto, & Pahnila, 2006; Liu & Guo, 2008). Here I inherited such definition of end users in the UNC Dropbox System testing and define the end user computing satisfaction as the affective attitude towards a specific information system by people who interact with the system directly. Such definition is consistent with and similar to some other researchers’ definitions on the End User Computing Satisfaction in the information system area (Igbaria & Nachman, 1990; Doll & Torkzadeh, 1988).

In 1988, Doll & Torkzadeh(1988) proposed a second order factor model for end-user computing satisfaction (EUCS) which contained five first order factors (content, accuracy, format, ease of use and timeliness) measured by twelve second order elements (4 2nd order elements measured content, 2 measured accuracy, 2 measured format, 2 measured ease of use and 2 measured timeliness). The twelve second order elements are actually twelve questions of which answers would directly reflect and elaborate users' opinions on their corresponding first order factors. Hence, the twelve second order elements can also be interpreted as independent variables which measure the five first order outcome variables. The combination of the results of the five outcome variables then leads to the actual end-user computing satisfaction level. At the initial stage of the research, Doll and Torkzadeh compiled 38 specific items with two global items from previous research to measure the EUCS construct. Adopting a five-point Likert-type scale, they pre-tested the instrument in a pilot study based on user self-reported data from a sample of 96 end users. 15 items were then eliminated by the correlations between corrected item total scores and item score. Another 5 items were also deleted because they were similar to other items with only slight difference in terms of wording. The remaining items went into the next round of factor analysis based on 618 users' responses. Consequently, only 12-items were retained in the instrument and became part of the EUCS instrument.

Doll double confirmed and improved his model in a confirmatory factor analysis of his previous research with new data collected from the same referent population in 1994

(Doll, Xia, & Torkzadeh, 1994). EUCS has then been used widely by many other Management of Information Systems (MIS) researchers to evaluate different kind of information systems like website (Abdinnour-Helm, Chaparro, & Farmer, 2005), online banking services (Pikkarainen, Pikkarainen, Karjaluoto, & Pahnila, 2006) and online shopping systems (Liu & Guo, 2008). Although slight modifications like presentation of items have been applied in these researches, the instrument itself has been validated multiple times to be psychometrically stable and reliable. Besides system context universality, EUCS also has been proved by Deng et al. to be culture independent (Deng, Doll, Al-Gahtani, Larsen, Pearson, & Raghunathan, 2008). Based on data collected from organizations of five nations/world regions (US, Western Europe, Saudi Arabia, India and Taiwan), they performed a multi-group invariance analysis and examined that EUCS provided equivalent measurement across cultures.

Based on the reviews of past literature above, the EUCS instrument, which has been validated for several times by different researchers, appears to be an appropriate tool to evaluate the success of the UNC Dropbox system in this project.

System Construction and Evaluation

Initial Preparation

Before starting to construct the UNC Dropbox System, my project advisor, Dr. Paul Jones, and I have met several times to discuss on the system requirements and specifications. We have made a consensus that the UNC Dropbox System would be built

based on the open source Dropbox System developed by University of Delaware (UD Dropbox System). While keeping the major functions of the original system, the UNC Dropbox System should adopt the Shibboleth authentication method using the UNC Single Sign-on service and the user interface of the original system should be modified to enhance system usability.

When the consensus was reached, I started to construct the system based on the requirements.

Server Configuration and the original system setup

The first thing I did was applying for a server running Ubuntu 10.04 LTS operating system from Ibiblio.org. I also requested sudo permission on the server so that I have enough authority to make necessary configurations. When the server was ready, I installed Apache2, PHP5, SQLite database and Postfix on the server and test to ensure they were working appropriately. The SSL module on the Apache 2 HTTP server was also enabled to support secure data communication. These were key components to make sure the open source UD Dropbox System can be set up on the server. When the initial configuration was done, I then downloaded the latest source package of the UD Dropbox System from its developer's website and put it in the web directory. By setting up some basic configurations on the system's configuration page and switched the system authentication to the static mode (testing mode), the UD Dropbox System was able to run on the server.

When I got the original UD Dropbox System set up on the server, I started to explore into details on how the system works and made notes on where modifications need to be applied. Once I was familiar enough with the original system, I began to read through its source code and tried to match its functions with backend source code. This process also helped me to locate the code sections that I need to make changes.

After all the previous work was done, I then started to make modifications on the system based on the pre-determined system requirements. The first modification I did on the system was its authentication method.

Authentication Method Modification

In order to modify the authentication method so that Shibboleth authentication can be performed in the system, the server needed to be configured to accommodate with the Shibboleth service provider, which was actually a client used to communicate with and receive authentication information from the Shibboleth Identity provider. In this project, the Shibboleth Identity provider was the UNC Single Sign-on server. I installed the Shibboleth package on the server and then fellow a tutorial on the Shibboleth 2 documentation website to make sure the service provider was correctly configured on my server. After that, I modified the Shibboleth service provider configuration file to make it able to talk with the UNC Single Sign-on server. When these initial settings were completed, the most challenging part of the modification started. The original UD Dropbox system used LDAP authentication with browser Cookies to differentiate users'

login status and permissions. In the Shibboleth authentication model, Cookie control from PHP scripts on the application server was infeasible because the whole authentication process was performed at the authentication server. According to this fact, I rewrote the old authentication method and configured the Apache 2 HTTP server to help enforce user session control. There are two kinds of session control in the Shibboleth authentication model, the active session control and the lazy session control. The active session control means all users who intend to access the protected page are required to be authenticated. The lazy session control means any user can access a protected page but some parts of the page are only available to authenticated users. PHP scripts could differentiate the user status by reading server variables stored in users' web browser. These variables were passed from the Shibboleth Identity provider when user got authenticated. I applied the active session control on the system function pages which were supposed to be used only by UNC authenticated users only and the lazy session control on the other pages which were both available to inside and outside users. All system pages were set to be controlled either by the active session or the lazy session except the system "About" page. Such design ensured that the whole system were under the protection of the Shibboleth authentication system and eliminated the possibility of unauthorized users to misuse unpermitted functions.

User Interface Upgrade

When the authentication method modification was done, I moved on to upgrade the user interface of the Dropbox system. The modifications included changing color theme

of the system to make it consistent with the UNC-CH style; adding welcome texts to remind user about their login status on every page; adding information to the file sharing page to remind users about the uploading limit; and implementing file uploading progress bar to give users a clear view of file uploading status while waiting for the server response. Among all these modifications, implementing file uploading progress bar was the most challenging task. I have researched many solutions on the internet to deal with this implementation. Most of the solutions required to use Adobe Flash or other programming languages like JSP, Perl or Python. The downside of using Adobe Flash solution was that end users would need to install Flash player on their web browsers in order to use the UNC Dropbox System. It was somewhat opposite to my initial intention on implementing this upgrade because I wished to enhance the usability of the system instead of brining users extra inconvenience. Moreover, using Flash on the system would also obstruct users with slow internet connections from using the system since it would take up lots of bandwidth to download the Flash contents. Progress bar solutions using other programming languages would increase maintenance difficulties to the system in future. I finally found a solution posted on the IBM developersWork website which utilized the PHP 5.2 with Alternative PHP Cache package and javascript to implement a dynamic progress bar. This solution required collaborations of three PHP scripts, the inner page, the parent page and the status tracking page. The basic operation logic is that the parent page hosts the inner page using the iframe tag. The inner page contains a form with input field for users to select files they want to upload. When users click on the form

submission button on the inner page, a hidden variable called APC_UPLOAD_PROGRESS with the value of a unique identification number will be sent to the server. The server will then store the file uploading data point in the cache during the uploading progress and using the unique identification number as the key to retrieve the data from the cache. As soon as users click on the submit button on the inner page, a javascript function on the parent page will be called and the unique identification will also be passed to that function. The purpose of the function is to display the uploading progress bar on the parent page and calling the status tracking page to read the cached uploading data recursively. Every time when the javascript fuction get response from the status tracking page, it will update the progress and percentage number on the uploading progress bar on the parent page. This solution really matched my requirements on the file uploading progress bar and did not bring extra inconvenience to end users. I then implemented and tested it on the file sharing page of the Dropbox system. It worked well.

Internal Testing and IRB Application

When I finished modifying the Dropbox system, I started to conduct internal testing and apply for the IRB approval for the close beta user testing. I invited my project advisor, the project coordinator and the project consultant from UNC-CH Information Technology Service to participate in the internal testing. The purpose of the internal testing was to test the system functions in different conditions to make sure they work appropriately and all pre-determined system requirements were achieved. The internal

testing feedbacks were quite satisfactory and the UNC Dropbox System was officially ready for close beta user testing.

User Testing Data Collection

As discussed with my project advisor, we decided to follow the typical software development procedures and divided the user testing into two separate phases. Phase one was a small group testing or close beta testing during which the system was set up on the testing server and was only open to a selected group of people with professional knowledge on system evaluation. The purpose of the phase one testing was to evaluate the overall performance of the system, look for major system bugs and collect improvement suggestions from experts. Phase two testing would be an open beta testing during which the system would be moved to a production server and open to the public. The purpose of the phase two testing was to evaluate the system performance under imitated production environment, check the request handling limit of the system and collect system traffic and usage data to estimate server configurations in the real production. Due to the time restriction, at current stage, I only conducted the phase one testing on the UNC Dropbox System. The phase two testing may be conducted after the submission of this report.

Since phase one testing is a small group testing and we want to collect more professional suggestions from system evaluation experts, we decided to limit the testing participants to twelve people and recruit them from three organizations (four people from

each) where most people got general knowledge on system testing. The three organizations were UNC School of Information and Library Science (SILS), Ibiblio.org and UNC Information Technology Services (ITS). To recruit testing participants from SILS, I sent out a recruiting email through the department administrative office to all students, faculties and staffs at SILS. People who were voluntary to join in the phase one user testing should reply the email to sign up. I received eight registration requests from SILS in total and random sampled four people for system evaluation. To recruit testing participants from Ibiblio.org, my project advisor, Dr. Paul Jones, help me send a recruiting email to the Ibiblio's mailing list. I received six registration requests from Ibiblio.org in total and four people were selected to join in the user testing. To recruit testing participants from ITS, our project consultant, Ethan Kromhout, who was also an employee at ITS, recommended four of his folks to participate the user testing. When I recruited enough participants, I sent them the study consent form together with a user testing instruction document to guide them what they should do and focus on during the user testing. Generally, they were asked to access the system; read through the about page to understand the characteristics and basic functions of the system; use and test whatever functions they were interested in or they thought bugs might occur; and fill in a survey designed based on the EUCS model to indicate their satisfactions. Every user had an average of two days to play with the system and submit the survey. The survey contained five sections and fifteen questions in total. The five sections corresponded to the five first-order factors in the EUCS model, which represented five key aspects of the system

to be evaluated (system content, accuracy, format, ease of use and timeliness). The fifteen questions except the first and the last were distributed in the five sections (2 to 4 questions in each section). Each question asked for users' opinions on a more detailed part of an aspect of the system. The combined results of all questions in one section represented users' satisfactions with the particular aspect of the system. The first and last questions were for general information collection purpose asking for users' affiliated organization name and system improvement suggestions. The 12th question was designed with my own interests to know the system response speed. The rest twelve questions were designed based on the twelve second order elements/questions in the EUCS model with minor presentation changes applied (for detailed design of the survey, please refer to Appendix A). The thirteen core questions in the five sections were all in the multiple choice format with five point Likert-scale options (from strongly disagree to strongly agree). Each question was companied with an optional comment field for testers to leave messages.

At the end of the phase one user testing, I received ten completed surveys (4 from SILS participants, 4 from Ibiblio.org participants and 2 from ITS participants), which represented an 83.3% response rate.

Evaluation Result Analysis

User Satisfaction on the UNC Dropbox System

To evaluate user satisfaction on the UNC Dropbox System, the first step is to quantify the collected survey results. The core of the user evaluation survey is the thirteen multiple choice questions with five point Likert-scale options in the five sections. I adopted a series of ordinal user satisfaction scores from 1 point to 5 points to indicate the five point ordinal Likert-scale options (strongly disagree=1, disagree=2, neutral=3, agree=4, and strongly agree=5). For example, if a user selected 'neutral' option as the answer for a question in the survey, the user satisfaction score of this question would be 3 points after quantification. According to the survey design, the higher score a question got from a user means the more satisfied the user is with a particular part of a particular aspect the system represented by the question. When I quantified all the user answers in collected user surveys, I then summarized user satisfaction scores from question, section and system perspective in a table (Table 1).

Strongly Agree=5
Agree=4
Neutral=3
Disagree=2
Strongly Disagree=1

UNC Dropbox System User Evaluation Survey Summary

			Mean Score		
			For question	For section	For system
UNC Dropbox System	Content	Question 2	4.4	4.28	4.25
		Question 3	4.2		
		Question 4	4.5		
		Question 5	4		
	Accuracy	Question 6	4.4	4.40	
		Question 7	4.4		
	Format	Question 8	4.3	4.15	
		Question 9	4		
	Ease of Use	Question 10	4.3	4.40	
		Question 11	4.5		
	Timeliness	Question 12	4.2	4.03	
		Question 13	4		
		Question 14	3.9		

Tabel 1: UNC Dropbox System User Evaluation Survey Summary

In Table 1, the mean user satisfaction scores for each survey question, each survey section and for the whole system are displayed in three separated columns. The mean user satisfaction score for each survey question was calculated by the arithmetic mean of all user assigned scores for the question. The mean satisfaction score for each survey section was calculated by the arithmetic mean of mean scores of questions contained in that section. And the mean satisfaction score for the whole system was calculated by the arithmetic mean of the mean scores of all five sections.

The UNC Dropbox System received an average user satisfaction score of 4.25 (max score 5), which meant testing participants were generally satisfied with the system performance and preliminarily indicated the success of the system. Exploring into more details, among the five evaluated aspects, testing participants were most satisfied with the accuracy and ease of use of the system, which both received 4.40 average user satisfaction score. This result indicated that almost all system functions worked accurately as user expected and the modified user interface did provide user with wonderful experience while using the system. As shown in the table, users were most unsatisfied with the timeliness of the system (4.03), which meant that the system sometimes either did not response quickly enough to user requests or did not provide information to user in time. According to users' individual answers, the three questions in the timeliness sections also received the most distributed user ratings among all questions. Some user only rated 1 point or 3 points to the questions while others assigned

4 points and 5 points. The possible reason for such a big gap between user experiences on the system timeliness may due to the different networks that users used to test the system. According to my own experience, testing the system at school using the campus network would receive system responses much faster than testing the system at my home using the Time Warner Cable network because the campus network had a faster uploading speed and the project and email server were both located on the campus network. Another possible reason is that system timeliness is an aspect that requires more subjective judgments from user to evaluate. Some people might define instant responses from the system as timeliness while others might think less than 30 seconds of waiting was acceptable. The content and format aspects of the system received average user satisfaction score of 4.28 and 4.15 respectively, which indicated that testing participants were generally satisfied with the two aspects but expected for more improvements.

Overall speaking, testing participants were quite satisfied with the performance of the UNC Dropbox System. All five key aspects of the system received a user satisfaction score higher than the satisfied level (4 points). Such positive feedbacks indicated the success of the system modification and implementation. However, testing participants also expressed their expectations on the future improvements of the system. Some of their suggestions did reveal several inconsiderate system designs and point out our future system development directions.

System Improvement Suggestions Summary

User suggestions were collected from the comments fields after each survey question and the answer of the last question in the survey. By reading through and filtering out several user comments which did not have concrete meanings on the system improvements, I grouped the remaining user suggestions into three categories, user interface improvement suggestions, system function improvement suggestions, and legal issue and system security suggestions.

User Interface Improvement Suggestions

Although testing participants were quite satisfied with the current UNC Dropbox system user interface and generously offered a high user satisfaction score, they still provided many useful suggestions to further improve the system usability. Starting from the system home page, one testing participant thought that the name of the system was quite misleading, which made him believed the UNC Dropbox System was a similar product to the commercial Dropbox.com at first glance. Though the two systems did share some similarities, they were ultimately different. He suggested changing the system name to a more unique and easy to remember identifier so that users would not be confused in future.

Now we move on to the system “About page”. This was the page received the most improvement suggestions. Some users complained that the help document was too long and too general. They suggested re-editing the document to make it more precise and specific to UNC users. They also suggested adopting a short graphic in the “About” page

to help user easily understand the file sharing and pick up process of the system. This is really a good idea to give users a brief instruction on how to use the system without the need to ask users to read long paragraphs.

Dropoff or file sharing page was another page that received several comments. Testing participants pointed out that the different units this page and the "About" page adopted to indicate the file uploading limits (in MB and GB) would easily make non-technical users to be confused. They suggested that using a unified representation to mark the file uploading limit would save the unit conversion time for users. Moreover, testing participants also discovered a minor bug in the file sharing summary page. They found that when the submitted note field for an uploading document contained a long text, the response file sharing summary page would not wrap the long text and therefore the text would run off the web page.

The above user interface improvement suggestions from testing participants were really helpful. Some of them are very hard to be noticed and thought about by developers. Implementing these suggestions only need a few efforts but the effect will be significant for the enhancement of user experience with the system.

System Function Improvement Suggestions

Testing participants have also offered several suggestions to further improve the UNC Dropbox system functions. The first suggestion was on extreme case error handling. One participant discovered that the system could not work appropriately while

users trying to upload zero size files and files with extremely long file name. In both cases, the file could be correctly uploaded but the notification email could not be delivered to recipients. This was an interesting bug which we did not notice during the internal user testing. Although such cases happened rarely in the real production environment, they still deserved attention from developers because user behaviors and intentions while using the system were almost impossible to be predicted. The second suggestion from testing participants was on Unicode support for uploading file names. The current system does not support uploaded file names in Unicode language. The file name cannot be correctly displayed in both the file sharing summary page and notification email. UNC-CH has many foreign students and departments which are doing foreign culture and language studies. They may have the needs share files in Unicode languages. Therefore, this is really an awesome suggestion expanding the capabilities of the UNC Dropbox System. The third suggestion from testing participants was to make the system support HTML 5 drag and drop feature while selecting sharing files. Implementation of this feature will enhance the user experience with the system and make the system look very cool. However, it should not be set as a short term goal for the system improvement since HTML 5 standards were still under debate and not all web browsers support this feature. The last testing participant suggestion on system functions was to implement an automatic email validation mechanism. Testing participants found out that currently the system did not validate neither the internal nor external email addresses of recipients. Users needed to check their input by themselves carefully to

ensure the recipients can receive the notification emails. They hoped that the system should have the capability to at least validate the internal email addresses.

Legal issue and System Security Suggestions

Testing participants expressed their worry on the legal issues and system security in their suggestions. The current system implementation does not restrict on the file types that users can share with others, which does bring great convenience to the end users. But at the same time, it also leads to challenges from legal and system security aspects. How to prevent users from sharing illegal or malicious contents like pirated software and computer virus using the system is a critical issue calling for solutions. In the future, file type restrictions may be imposed for outside users using the UNC Dropbox system. But it is not guaranteed that all inside users will self-regulate to follow the rules. For detailed process on handling this issue, we still need more internal discussions. We will try to come out with a solution to deal with this issue while not sacrifice the convenience of the system.

Future Improvements

Based on the collected user suggestion, we can foresee the future improvements on the UNC Dropbox System from short term and long term perspective.

The short term improvements on the system should be both cost-effective and easy to be implemented. They are planned be available on the system before the second phase of user testing. For the system user interface, I plan to re-design the system “About” page to

make it more precise and specific to UNC-CH users and adopt simple graphics to describe the dropoff and pick up process. I also plan to unify presentation of the file upload size limit on different pages. In addition, the word-wrap function will also be implemented on the dropoff summary page to avoid the out-of-page display problem caused by long note text. For the system functions, I plan to solve the handling problems on zero size file upload and long name file upload. Moreover, the Unicode file name support and internal email address validation mechanism are also expected to be implemented depending on the development progress.

For the long term improvements on the system, the focus will be on solving the legal and security issues with the system and exploring to realize the HTML 5 features on the UNC Dropbox System.

Conclusion

In this project, I have designed, modified and implemented an online file sharing system dedicated for the UNC-CH affiliated members to use. The system provided UNC-CH users a platform to share files with other people without the need to worry about the email attachment size limit, and their course, department or project enrollment status. A small group user testing has been conducted to test user satisfactions with the performance of the system. Testing participants were selected from three organizations where people have professional knowledge on system testing. The survey analysis results

indicated that testing participants were quite satisfied with the UNC Dropbox System from five key aspects (system content, accuracy, format, ease of use and timeliness). Besides expressing their satisfactory with the system, testing participants also offered many helpful suggestions which will serve as guidelines for the future improvements of the UNC Dropbox System.

References

Abdinnour-Helm, S. F., Chaparro, B. S., & Farmer, S. M. (2005). Using the End-User Computing Satisfaction (ECUS) to Measure Satisfaction with a Website. *Decision Science , 36* (2), 341-364.

Baroudi, J., & Orlikowski, W. (1988). A Short From Measure of User Information Satisfaction: A Psychometric Evaluation and Notes on Use. *Journal of Management Information Systems , 4* (4), 45-59.

Barret, V. (2010, 6 28). *Dropbox: Files Without Borders.* Retrieved 9 16, 2010, from Forbes: http://www.forbes.com/forbes/2010/0628/technology-file-sharing-dropbox-google-files-without-borders.html?boxes=Homepagechannels

Deng, X., Doll, W. J., Al-Gahtani, S. S., Larsen, T. J., Pearson, J. M., & Raghunathan, T. (2008). A Cross-Cultural Analysis of the End-User Computing Satisfaction Instrument: A Multi-Group Invariance Analysis. *Information and Management* , 211-220.

Doll, W. J., & Torkzadeh, G. (1988, June). The Measurement of End User Computing Satisfaction. *MIS Quarterly* , 259-274.

Doll, W. J., Xia, W., & Torkzadeh, G. (1994, December). A Confirmatory Factor Analysis of End User Computing Satisfaction Instrument. *MIS Quarterly* , 453-461.

Ein-Dor, P., & Segev, E. (1982). Organizational Context and MIS Structure: Some Empirical Evidence. *MIS Quarterly , 6* (3), 55-68.

Ginzberg, M. (1978). Finding an Adequate Measure of OR/MS Effectiveness. *Interfaces , 8* (4), 59-62.

Igbaria, M., & Nachman, S. A. (1990). Correlates of user satisfaction with end user computing. *Information & Management* , 73-82.

Igbaria, M., Pavri, F., & Huff, S. (1989). Microcomputer Applications: An Empirical Look at Usage. *Information & Management , 16* (4), 373-388.

Ives, B., & Baroudi, J. (1983). The Measurement of User Information Satisfaction. *Communication of the ACM , 26* (10), 785-793.

January 2011 Web Server Survey. (2011, 1 12). Retrieved 3 22, 2011, from NetCraft: http://news.netcraft.com/archives/2011/01/12/january-2011-web-server-survey-4.html

Lefkovits, H. (1979). A Status Report on the Activities of Codasyl End-User Facilities Committee (EUFC). *Information and Management* , 137-163.

Liu, C.-T., & Guo, Y. (2008). Validating the End-User Computing Satisfaction Instrument for Online Shopping Systems. *Journal of Organizational and End User Computing* , 74-96.

Lucas, H. (1978). Empirical Evidence for a Descriptive Model of Implementation. *MIS Quarterly , 2* (2), 27-41.

Martin, J. (1982). *Application Development Without Programmers.* Englewood Cliffs, NJ: Prentice Hall, Inc.

McLean, E. (1979). End-Users as Application Developers. *MIS Quarterly* , 37-46.

Pikkarainen, K., Pikkarainen, T., Karjaluoto, H., & Pahnila, S. (2006). The Measurement of End-User Computing Satisfaction of Online Banking Services: Empirical Evidence from Finland. *International Journal of Bank Marketing , 24* (3), 158 - 172.

Powers, R., & Dickson, G. (1973). MIS Project Management: Myths, Opinions, and Reality. *California Management Review , 15* (3), 147-156.

Rockart, J., & Flannery, L. (1983). The Management of End User Computing. *Communication of the ACM* , 776-784.

S.Rivard, & Huff, S. (1988). Factors of Success for End-User Computing. *Communications of the ACM , 31* (5), 552-561.

Schwartz, A. (2007). UD Dropbox 2.0: Collaboration Magic. *SIGUCCS* (pp. 305-309). Orlando: ACM.

Tait, P., & Vessey, I. (1988). The Effects of User Involvement on System Success: A Contingenc Approach. *MIS Quarterly , 12* (1), 91-108.

Wauters, R. (2010, 1 20). *Dropbox Announces 4 Million Users, Hires A VP From Salesforce.* Retrieved 9 16, 2010, from TechCrunch: http://techcrunch.com/2010/01/20/dropbox-4-million-user/

Appendix A: UNC Dropbox System User Testing Survey

For each of the following statement about your experience with the UNC Dropbox system, please indicate your satisfaction level.

Please notice that you can only select one answer for each statement.

You can write down any comments and suggestions for the system on the fields below any question or at the end of the survey.

1. Please indicate your affiliated organization

_____

Section I: System content

2. Did the UNC Dropbox system provide the precise information you need to operate it?
    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

    Comments:_____

3. Did the UNC Dropbox system provide help that seemed to be just about exactly what you need?
    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

    Comments:_____

4. Did the UNC Dropbox system provide sufficient information for you to use it?

    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

Comments:_____

5. Did the information ('about' page, function labels, dropoff & pickup reports, error messages etc.) provided by the UNC Dropbox system meet your need?

    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

    Comments:_____

Section II: System accuracy

6. Was the UNC Dropbox system accurate (all functions worked as you expected)?

    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

    Comments:_____

7. Were you satisfied with the accuracy of the UNC Dropbox system?
    1. Very Dissatisfied
    2. Dissatisfied
    3. Neutral
    4. Satisfied
    5. Very Satisfied

    Comments:_____

Section III: System format

8. Did you think the UNC Dropbox system present information in a useful format?

    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

Comments:_____

9. Did the UNC Dropbox system present information clearly?
    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

Comments:_____

Section IV: System ease of use

10. Was the UNC Dropbox system user friendly?

    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

Comments:_____

11. Was the UNC Dropbox system easy to use?
    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

Comments:_____

Section V: System timeliness

12. Did the UNC Dropbox system response quickly?

    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

Comments:_____

13. Did you get information you need in time?
    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

    Comments:_____

14. Did the UNC Dropbox system provide up-to-date information?
    1. Strongly Disagree
    2. Disagree
    3. Neutral
    4. Agree
    5. Strongly Agree

    Comments:_____

Section VI: Additional Suggestions

15. **If you have any more suggestions for us to improve the UNC Dropbox System, please write down:**

    _____

    _____

    _____

    _____

Appendix B: UNC Dropbox System Survey Results (Quantified)

Strongly Agree=5		Question 2	Question 3	Question 4	Question 5		Question 6	Question 7
Agree=4		5	5	5	5		5	5
Neutral=3		4	3	4	2		4	4
Disagree=2		5	5	5	5		5	5
Strongly Disagree=1		5	5	5	5		5	5
	Content	4	4	5	4	Accuracy	3	4
		4	3	5	2		4	4
		4	4	4	4		4	4
		5	4	4	5		5	5
		4	4	4	4		5	5
		4	5	4	4		4	3
	Average	4.4	4.2	4.5	4	Average	4.4	4.4
		Question 8	Question 9		Question 10	Question 11		
		5	5		5	5		
		4	3		4	4		
		5	5		5	5		
		5	5		5	5		
	Format	4	4	Ease of Use	4	4		
		4	2		4	5		
		4	4		4	4		
		4	4		4	4		
		3	4		4	4		
		5	4		4	5		
	Average	4.3	4	Average	4.3	4.5		
		Question 12 (Non-EUCS)	Question 13	Question 14				
		5	5	5				
		1	1	1				
		5	5	5				
		5	5	5				
	Timeliness	4	3	4				
		5	3	3				
		4	4	4				
		4	5	5				
		5	4	4				
		4	5	3				
	Average	4.2	4	3.9				